Method of Scalable Non-Blocking Shared Memory Output-Buffered Switching Of Variable Length Data Packets From Pluralities Of Ports At Full Line Rate, And Apparatus Therefor

<u>Field</u>

The present invention relates to communication data switching between pluralities of input and output ports, and, more particularly, to problems and limitations of present - day generally input-buffering system architectures and the like for the switching of variable-length data packets –limitations in the available number of ports for current data switching "speeds" and "feeds"; limitations with current data transmission delays, and in current available quality of service, including multiplexing jitter, interruptions, and in bandwidth, latency guarantees for particular data transmission services, and in obviating deleterious head-of-the-line blocking and non-scalability of architecture.

The usual "feed" today is 8 to 12 ports, but this can go up to as time goes by. The "speed" today is, say, OC192 (which is 10 gigabytes), but it can also go to OC768 which is 40 gigabytes, and then beyond.

<u>Background</u>

Prevalent products in the industry today can only support 8 to 12 OC192 ports, and they suffer from the other limitations mentioned above.

To endeavor to meet some of the quality of service requirements concurrently with data "speed" and "feed" requirements, the prior art has most commonly taken the before-described input buffering approach, wherein the input data is locally buffered on an input port that has no "knowledge" of what input data may also be present at other

input ports and contending for the same output port destination. The input port merely blindly makes the request of the input buffered switch to direct its data to the particular output port; and this prior architecture thus has had to live with its classic problems of potential head-of-the line (HOL) blocking and inability to guarantee delay and jitter in quality of service. The input-buffered systems, accordingly, have to put up with sometimes even unrealistic periods of time before data can make its way to the switch for enabling transmission to destination output ports.

The particular output-buffered approach of the invention, on the other hand, uses a central shared memory architecture comprised of a plurality of similar successive data memory channels defining a memory space, with fixed limited times of data distribution from the input ports successively into the successive memory cells of the successive memory channels, and in striped fashion across the memory space. This enables non-blocking shared memory output-buffered data switching, with the data stored across the memory channels uniformly. By so limiting the time of storing data from an input port in each successive memory channel, the problem is admirably solved of guaranteeing that data is written into memory in a non-blocking fashion across the memory space with bounded delay.

Summary of Invention

From one of its important viewpoints, accordingly, the invention embraces a method of receiving and outputting a plurality m of queues of data traffic streams to be switched from data traffic line card input ports to output ports, that comprises, providing a plurality n of similar successive data memory channels each having a number of

2

memory cells defining a shared memory space assigned to the m queues; providing

buffering for m memory cells, in front of each memory channel to receive and buffer data

switched thereto from line card traffic streams, and providing sufficient buffering to

absorb a burst from up to n line cards; and distributing successive data in each of the

queues during fixed limited times only to corresponding successive cells of each of the

successive memory channels and in striped fashion across the memory space, thereby

providing non-blocking shared memory output-buffered data switching.

Preferred and best mode embodiments and architectural design features are

hereinafter more fully detailed.

Drawings

The invention will now be described in connection with the accompanying

drawings, Fig. 1 of which is a combined generalized block and circuit diagram of a

preferred architecture for practicing the data write-path method of the invention; and

Fig. 2 is a similar diagram of read-out from the shared memory channel system of

Fig. 1.

Preferred Embodiment(s) Of The Invention

Referring to Fig. 1, an illustrative preferred memory architecture for practicing

the invention is shown having, for the write path, a plurality n of similar successive data

memory channels or banks (say, for 256 megabytes times n storage channels), labeled

Memory Channel $_o$ through memory channel$_{n-1}$ for storing and outputting m queues of

variable length data traffic streams Queue$_o$ through Queue$_{m-1}$ from respective data traffic

line cards Line Card$_o$ through Line Card$_{n-1}$ at input ports I, with, say, 10 Gigabits/sec.of

bandwidth, and stored in the memory channels. Each of the n data memory channels is provided with a buffer having m memory cells, with the memory channels defining a shared memory space assigned to the m queues. The buffers are shown connected in front of each memory channel and are illustrated as in the form of first-in-first-out buffers $FIFO_0$, $FIFO_1$,... $FIFO_{n-1}$, to receive and buffer data switched thereto at SW from the line cards. In accordance with the invention, the buffers are designed to provide sufficient buffering to absorb a burst of data from up to n line cards; i.e. big enough to store data for m cells and to absorb a burst of, for example, OC192 traffic of variable length data packets from the line cards at the input ports I. [Example: 64 OC192 or 16 OC768 ports.] The maximum depth of each FIFO at the front of each memory channel is thus made equal to the number m of queues in the system.

Further in accordance with the invention, the data of the variable-length queues is applied or distributed only for fixed limited time(s) to corresponding successive cells of each of the successive memory channels so as to distribute these time-bounded inputs in striped fashion across the memory space of the channels. Within each period, every memory channel or bank receives data in about the same number of data cells, though arrival time is traffic dependent; and this, whether there is a data burst or the data is distributed equally throughout the period.

Two exemplary (and extreme condition) traffic scenarios may be considered. In the first, all traffic streams from the line cards may be destined to one queue. Since the cell addresses are assigned continually, all the memory channels will absorb a data burst. There will be no accumulation in any FIFO, provided the aggregation of bandwidth to memory is made to match the input bandwidth.

In a second extreme scenario, all the cells may happen to end on the same memory channel. The FIFO at the front of that memory channel will absorb the burst; and the next burst to come along, will move to the next memory channel.

This demonstrates that with the proper sizing of the FIFOs to absorb any data burst at the front of each memory channel, the burst problem is well solved and with a bounded latency. As above explained, moreover, the depth of the FIFOs is set at about the number of queues supported by the system, and the aggregated bandwidth between the FIFOs in the memory channels is adjusted, as indicated previously, at least to match the input bandwidth.

Through the invention, accordingly, not only is non-blocking shared memory output-buffered data switched, but the bandwidth can be assigned and guaranteed to designated users. If a predetermined assigned depth is exceeded by a user, such excess is stored in available unoccupied shared memory and may be additionally charged for, to that user.

Fig 2 illustrates the read path architecture of the invention for use with the write path system of Fig. 1, providing for every line card, a corresponding FIFO that is able to draw from the shared memory and at the full bandwidth of the shared memory in a TDM type fashion. In the read operation, it is important that the bandwidths are completely balanced to each line card with equal access to the shared memory system, wherein each line card gets its fixed limited time slot to read out the required amount of data to satisfy the bandwidth needs. As an example, Line Card$_o$ and FIFO$_o$ of Fig. 2 read from the shared memory the full bandwidth of the shared memory going up the FIFO$_o$; the Line Card, and its corresponding FIFO, will get its share of the full bandwidth from the shared

memory, and so on--each line card getting each required share of the shared memory bank data.

Further modifications will occur to those skilled in this art, and such are considered to fall within the spirit and scope of the invention as defined in the appended claims.